

Sound Exchanges With Voice Service Systems

Field of the Invention

The present invention relates to sound exchanges with voice service systems.

Background of the Invention

In recent years there has been an explosion in the number of services available over the World Wide Web on the public internet (generally referred to as the "web"), the web being composed of a myriad of pages linked together by hyperlinks and delivered by servers on request using the HTTP protocol. Each page comprises content marked up with tags to enable the receiving application (typically a GUI browser) to render the page content in the manner intended by the page author; the markup language used for standard web pages is HTML (Hyper Text Markup Language).

However, today far more people have access to a telephone than have access to a computer with an Internet connection. Sales of cellphones are outstripping PC sales so that many people have already or soon will have a phone within reach where ever they go. As a result, there is increasing interest in being able to access web-based services from phones. 'Voice Browsers' offer the promise of allowing everyone to access web-based services from any phone, making it practical to access the Web any time and any where, whether at home, on the move, or at work.

Human-to-human sound interaction is, in fact, far richer than the simple dialogs currently possible through the use of scripted voice service systems.

It is an object of the present invention to provide a method and system which enhances the available forms of sound interaction with a voice service system.

Summary of the Invention

According to one aspect of the present invention, there is provided a method of interacting with a human user through a sound service system, wherein the service system participates with the human user both in normal voice dialog exchanges, and in a multi-turn sound

exchange the form and content of which are pre-specified and already public, this sound exchange involving one or more cycles in each of which the service and user take turns to provide a noise or utterance with the appropriate pre-specified content.

- 5 According to another aspect of the present invention, there is provided a method of interacting with a human user through a sound service system, wherein the service system participates in a multi-turn sound exchange with the user, this sound exchange involving one or more cycles in each of which the service and user take turns to provide a noise or utterance the form or content of which is already public.

10

According to a further aspect of the present invention, there is provided a sound service system comprising a sound input channel for receiving and interpreting sound input signals, a sound output channel for generating sound output signals, and a dialog manager connected to an output of the sound input channel and an input of the sound output channel, the dialog manager being operative to manage the participation of the service system in exchanges with a user and comprising:

15

- means for managing participation of the service system in normal voice dialog exchanges with the user, and
- means for managing participation of the service system in a multi-turn sound exchange with the user, the form and content of this exchange being pre-specified and already public, and the exchange involving one or more cycles in each of which the service and user take turns to provide a noise or utterance with the appropriate pre-specified content.

20

- 25 According to a still further aspect of the present invention, there is provided a A sound service system comprising a sound input channel for receiving and interpreting sound input signals, a sound output channel for generating sound output signals, and a dialog manager connected to an output of the sound input channel and an input of the sound output channel, the dialog manager being operative to manage the participation of the service system in exchanges with a user and comprising:

30

- means for managing participation of the service system in normal voice dialog exchanges with the user, and

- means for managing participation of the service system in a multi-turn sound exchange with the user, the form and content of this exchange being pre-specified and already public, and the exchange involving one or more cycles in each of which the service and user take turns to provide a noise or utterance with the appropriate pre-specified content.

Brief Description of the Drawings

A method and system embodying the invention, for multi-turn sound exchanges with a sound service system, will now be described, by way of non-limiting example, with reference to the accompanying diagrammatic drawings, in which:

- . **Figure 1** is a diagram of a sound service system for effecting multi-turn dialog exchanges;
- . **Figure 2** is a diagram showing the sounds exchanged between a human user and the service system in respect of a first multi-turn sound exchange;
- . **Figure 3** is a diagram showing the sounds exchanged between a human user and the service system in respect of a second multi-turn sound exchange;
- . **Figure 4** is a diagram showing the sounds exchanged between a human user and the service system in respect of a third multi-turn sound exchange; and
- . **Figure 5** is a diagram showing a voice browser system including both a voice dialog manager and a multi-turn dialog manager.

Best Mode of Carrying Out the Invention

Figure 1 shows a sound service system for participating in a multi-turn sound exchange with a human user, this sound exchange involving one or more cycles in each of which the service and user take turns to provide a noise or utterance. Generally the form or content of the noise or utterance will already be public and known to the user so that they can participate fully in the exchange and gain a feeling of involvement.

The Figure 1 service system comprises a sound input channel 11 for receiving and interpreting sound inputs, a sound output channel 12 for generating sound output, and a multi-turn dialog manager 10 connected to the output side of the sound input channel 11 and the input side of the sound output channel 12. The sound input channel comprises a

microphone 13 feeding a speech recogniser 14 and a distinctive sound detection unit 15, this latter unit being designed to recognise specific non-word sounds such as handclaps and whistles. The sound output channel 12 comprises a distinctive sound generator 16 for generating non-word sounds such as handclaps and whistles, a text-to-speech converter 17, an audio server 18 for outputting pre-recorded sound segments, and loudspeaker 19 for receiving the outputs of the generator 16, converter 17 and server 18 and generating corresponding sounds. The multi-turn dialog manager 10 is operative to manage the participation of the service system in a multi-turn sound exchange, commanding the units of the output channel to generate appropriate sounds during the systems turn's in the multi-turn dialog and using the input channel to check for appropriate responses from the user. The multi-turn dialog manager 10 is, for example, arranged to interpret script files that define respective multi-turn dialogs.

Figure 2 illustrates a first multi-turn dialog, in this case initiated by the manager 10. The dialog proceeds as follows:

Service: "<XXX product tune> + <whistle>"
 Human: "<whistles back>"
 Service: "<Clapping sound>"
 Human: "<Claps>"
 Service: "XXX makes the best MP3 players<advertising>"

The distinctive sound generator 16 is used to generate the whistle and clap for the service system turns whilst the distinctive sound detection 15 unit detects these sounds repeated back by the human user. The TTS converter 17 is used for generating the spoken words "XXX makes the best MP3 players". The audio server 18 is used to generate the XXX product tune and possibly also the advertising material (though this could be scripted for the TTS converter).

Figure 3 depicts a wholly speech-based multi-turn dialog running as follows:

Service: "One , Two, Three.."
 Human: "Speak to me!"
 Service: "And now its four.."
 Human: "So tell me more!"

Service: <enters a normal dialog mode>

The multi-turn dialog actually terminates after the second human response. In the present case, the service system now changes to a normal voice dialog mode under the control of a voice dialog manager which can either be combined into one functional block with the multi-turn dialog (MTD) manager 10, or embodied in a separate functional block (not shown in Figure 1 but to be more fully described hereinafter with respect to the embodiment of Figure 5).

The multi-turn dialogs illustrated in Figures 2 and 3 were both initiated by the MTD manager 10 (for example, upon the user first contacting the service system). However, the user may also initiate the dialog by uttering or otherwise generating the first sound elements of the first turn of the dialog, this first turn being, now, a user turn. In this case, the MTD manager 10 recognizes these first sound elements and participates in the corresponding multi-turn dialog.

Figure 4 depicts a multi-turn dialog started by the user saying the word "wozsaar!" (in other words: "What's up" meaning "What is happening?"). This is repeated back by the service and this cycle the loops (repeats) either for as long as the user is willing to continue or until a timeout period, or turn count, expires. In the illustrated case, the user ends the multi-turn dialog by uttering an exit key word causing the service system to drop into a normal voice dialog mode.

Figure 5 shows a voice browser 50 provided with multi-turn dialog capability. Voice browser 50 is located in the communications infrastructure (being, for example, provided by a PSTN or PLMN operator or by an ISP). A voice browser allows people to access the Web using speech and is interposed between equipment 41 of a user 42, and a voice page server 40. This server 40 holds voice service pages (text pages) that are marked-up with tags of a voice-related markup language (or languages). When a normal voice dialog page (such as page 26) is requested by the user 42, it is interpreted at a top level (dialog level) by a voice dialog manager 22 of the voice browser 33 and output intended for the user is passed in text form to the output channel 12 of the browser. The output channel is here shown as comprising a language generator 30 for driving a Text-To-Speech (TTS)

converter 17 and a distinctive sound generator 16. The output of channel 12 is passed over a sound connection (such as a telephone voice circuit or VoIP connection) to the user equipment 41. User voice (and other sound) input is passed back over this connection to an input channel 11 of the voice browser. The input channel in this case comprises a speech
 5 recognition unit 14 and a distinctive sound detection unit 15, both feeding a language understanding unit 21. The input channel 11 uses lexicon and grammar data 25 to determine what sounds/words have been received and may seek to understand this input in the context of what has gone before. For normal voice dialog operation, the output of the channel 11 is passed to the voice dialog manager 22 which then determines what action is
 10 to be taken according to the received input and the directions in the original page.

The voice browser 50 further comprises a MTD manager 23 that is a distinct functional element from the normal voice dialog manager 22, the MTD manager operating according to one or more multi-turn dialog scripts 23. The scripts are, for example, loaded into the
 15 MTD manager 23 when the voice browser first contacts a voice website hosted on server 40, these scripts being retained whilst the voice browser remains at the same voice website (in contrast, as the browser moves between pages of the website, different scripts 26 are loaded into and out of the voice dialog manager 22).

20 The voice browser can operate in two modes, namely a normal voice dialog mode in which the voice dialog manager 22 is in control and runs its currently loaded script 26, and an MTD mode in which the MTD manager 23 is in control and runs a selected one of its currently loaded scripts 27. The current mode of the browser is held by mode unit 24 which indicates the current mode of the voice browser to the managers 22, 23, the language
 25 understanding unit, and the language generator 30.

When the browser is in its normal voice dialog mode, the language understanding unit 21 uses the grammar appropriate to script 26 but, in addition, is caused to look out for user input sound elements that correspond to the initial elements of any of the multi-turn dialog
 30 script starting with a user turn. If the initial elements of such a MTD script are detected in the input sound stream, the unit 21 changes the mode of the browser held by mode unit 24 to the MTD mode. As a result, MTD manager 23 assumes control of the following sound

exchange which it controls in accordance with the script 27 whose initial elements were detected.

The MTD mode can also be entered by the current script 26 requesting the unit 24 to
 5 change modes, the script 26 also informing the manager 23 which multi-turn dialog script 27 is to be executed.

When the browser is in its MTD mode, the language understanding unit 21 uses the grammar appropriate to the selected script 27 but, in addition, is caused to look out for
 10 user input sound elements that correspond to exit key words or phrases indicating that the user wishes to terminate the current multi-turn dialog exchange. If an exit key word or phrase is detected in the input sound stream, the unit 21 changes the mode of the browser held by mode unit 24 to the normal voice dialog mode. As a result, voice dialog manager 22 assumes control of the following sound exchange in accordance with script 26.

15 The normal voice dialog mode can also be entered as a result of the MTD manager causing the unit 24 to change the set mode upon the MTD manager finishing execution of a current multi-turn dialog script 27.

20 During the course of execution of a multi-turn dialog script, the user's input can be treated in a number of different ways:

- (a) - checked only for occurrence of a user input, not necessarily as expected;
- (b) - checked for expected user input;
- (c) - checked for one of several possible expected user inputs.

25 In respect of (b), if the expected user input is not received, the multi-turn dialog can be terminated or a correction dialog entered. In the case of (c), the identity of received input (if one of the expected inputs) can be used to determine which of several branches in the dialog is pursued by the MTD manager 23; alternatively, the identity of the received input can be used to select a particular voice dialog script 26 to be used when the normal voice
 30 dialog mode is entered at the end of the current multi-turn dialog, this identity being passed to manager 22; generally, however, the multi-turn dialog will serve no function in respect of accessing or controlling the course of the normal dialog exchanges.

Many variants are, of course, possible to the arrangements described above. For example, the MTD manager can be provided with functionality for ensuring that the service system executes its next turn promptly on the conclusion of the user's turn. This functionality can
5 include means for predicting when the user's input will terminate having regard to its speed of delivery. Such promptness of response increases the user's feeling of interaction with the service system.

Whilst the turns of the multi-turn dialogs will generally be public knowledge (that is, their
10 form and/or content are pre-specified and not confidential), this is not always necessary to achieve a feeling of involvement. In the preferred embodiments, the multi-turn dialogs are promotional in nature, promoting a commercial enterprise and/or its goods and/or its services. The multi-turn dialogs are not, by their very nature of being public, password exchanges nor are they merely formal greetings because the user's response to a standard
15 greeting phrase from the service (such as "Good Morning! How are you?") can be almost anything.

It will be appreciated by persons skilled in the art that the managers 22 and 23 will normally be implements in software and can, in fact, simply be separate instantiations of
20 the same manager class. Alternatively, the managers can be implemented as different methods of a general dialog manager object, or different software functions called as required by a controlling program.